

WHAT IS CLAIMED IS:

- 1 1. A data backup system for use with a server running a storage
2 application that writes and read data blocks to and from a self archiving log
3 structured volume, the data backup system comprising:
4 a self archiving log structured volume for copying blocks from
5 primary storage to backing storage while the storage application is running;
6 an agent to indicate to the self archiving log structured volume when
7 the blocks of the self archiving log structured volume are in a consistent state with
8 respect to the storage application; and
9 primary and backup storage operable with the self archiving log
10 structured volume to store the logged blocks and synch events of the self archiving
11 log structured volume.
- 1 2. The data backup system of claim 1 wherein:
2 the self archiving log structured volume includes a log having a
3 plurality of log segments and an index, the log segments including a current log
4 segment and active log segments for storing blocks of an active volume, wherein
5 the index shows the current position of each block in the log segments.
- 1 3. The data backup system of claim 2 wherein:
2 the self archiving log structured volume satisfies write block
3 requests from the storage application by copying the written block to the end of
4 the log and then updating the index with the current position of that block in the
log.
- 1 4. The data backup system of claim 2 wherein:

2 the self archiving log structured volume satisfies a read block
3 request by looking up the requested block in the index and then copying the
4 requested block from the log.

1 5. The data backup system of claim 2 wherein:
2 the self archiving log structured volume records a synch event by
3 writing a special block to the log containing the date, time, and other information
4 describing the synch event.

1 6. The data backup system of claim 2 wherein:
2 the self archiving log structured volume writes blocks to the current
3 segment of the log and then converts the current segment to an active segment
4 when the current segment is full.

1 7. The data backup system of claim 6 wherein:
2 the active segments of the log structured volume are in a time
3 sequential order.

1 8. The data backup system of claim 2 wherein:
2 the segments of the log are of equal size.

1 9. The data backup system of claim 2 wherein:
2 the log further includes a plurality of inactive segments and a
3 plurality of recycle segments.

1 10. The data backup system of claim 2 wherein:

2 the self archiving log structured volume is operable to construct a
3 volume index from any synch point by scanning backward in the log and updating
4 the index entry for each block to the most recent position in the log.

1 11. The data backup system of claim 10 wherein:
2 the current segment and the active segments of the log make up a
3 working set of segments having a fixed length from which all write request blocks
4 are satisfied.

1 12. The data backup system of claim 11 wherein:
2 the maximum length of the scan is the length of the working set of
3 segments and one additional segment.

1 13. The data backup system of claim 12 wherein:
2 the oldest active segment is designated as an inactive segment when
3 the working set of segments becomes full.

1 14. The data backup system of claim 13 wherein:
2 the inactive segments are compressed and archived by the self
3 archiving log structured volume to the backing storage.

1 15. The data backup system of claim 14 wherein:
2 the contents of an inactive segment on the primary storage are
3 discarded after being compressed and archived to the backing storage and the
4 inactive segment then becomes a recycle segment.

1 16. The data backup system of claim 2 wherein:

2 the size of each segment and the number of segments is determined
3 by policy.

1 17. The data backup system of claim 2 wherein:
2 the self archiving log structured volume is operable to migrate
3 inactive segments of the log to the backing storage.

1 18. The data backup system of claim 17 wherein:
2 the self archiving log structured volume is operable to ensure that a
3 volume can be reconstructed from a fixed number of log segments.

1 19. A method for backing up data using a self archiving log
2 structured volume having a sequence of numbered blocks of fixed length and a log
3 for recording log entries of write events and synch events to the volume, wherein
4 the log includes synch event entries, active log entries, and inactive log entries, the
5 method comprising:
6 performing a write event by copying data to a sequence of the blocks
7 of the volume, wherein each write event entry includes the number of the block
8 being written and the contents of the block being transferred;
9 performing a synch event to record a point in time when the blocks
10 of the volume were known to be in a consistent state, wherein each synch event
11 entry includes the time of the synch event;
12 performing a log entry for a write event, wherein a log entry for a
13 write event is an active log entry until superseded by a later write event for the
14 same block number and then the active log entry becomes an inactive log entry,
15 wherein when an active segment consists only of inactive entries and synch entries
16 the active segment becomes an inactive segment eligible to be archived;

00-010-DSX
STK 00010 PUS1

17 archiving an inactive segment to backing storage;
18 converting the archived inactive segment to a recycle segment
19 containing no entries and eligible to become the new current segment, wherein the
20 current log segment and the active log segment define a finite reconstruction length
21 for the volume;
22 recording index entries in an index of the log, each index entry
23 associated with a volume block number and a corresponding log entry; and
24 satisfying write events by adding an entry for a block of the volume
25 to the current log segment and updating the index entry for the block number with
26 the log location of the new log entry.

1 20. The method of claim 19 wherein:
2 inactive log entries are archived to backing storage without
3 knowledge of a storage application owning the volume because the log organization
4 eliminates the possibility of contention for the same blocks between the storage
5 application and the archiving process.